Redundancy Method for Critical Services in RIBF Control System

Akito Uchiyama #,A), Misaki KomiyamaB), Masayuki KaseB)
A) SHI Accelerator Service, Ltd. (SAS)
1-17-6 Osaki, Shinagawa-ku, Tokyo, 141-0032
B) RIKEN Nishina Center
2-1 Hirosawa, Wako-shi, Saitama, 351-0198

Abstract

In the RIBF (RIKEN RI Beam Factory) control system based on EPICS (Experimental Physics and Industrial Control System), there are IOCs (Input/Output Controller) to control various types of devices. Some of them are Linux IOCs and the others are VxWorks IOCs. All of Linux IOCs mount the directory which has EPICS programs with NFS (Network file system) service. The VxWorks IOCs use the FTP (File transfer protocol) for network booting and EPICS programs. Beside, the other critical server, such as the EPICS client and operation log system, relies on the centralized system. Therefore, System-wide failure occurs if one of them has a serious problem. To avoid this situation, we constructed Linux clusters with redundancy design based on High Availability system and DNS round robin.

RIBF 制御系におけるクリティカルなサービスの冗長化手法

1. はじめに

理研 RIBF の加速器制御システムは主に EPICS を用いた分散制御システムで構築されている[1]。分散制御システムの大きな長所として、部分的にトラブルを抱えたとしてもシステム全体が停止する事が無い、という事が挙げられる。我々のシステムでは EPICS IOC は分散化されているが、一方で EPICS CA (Channel Access) クライアントや EPICS データベースの運用等に集中型システムも利用している。それらは少ないマンパワーで効率良いシステム開発、メンテナンスが行えるという半面、IOC 以外のオペレーションに必須なサービスを集中型システムに依存している為に分散制御システムのメリットを完全に失っていた。以上の問題を解決する為に、RIBF 制御系で用いられているクリティカルなサービスを冗長化し、高可用性の確保を行った。

2. 障害発生時の被害

重要なサービスが停止して加速器制御が行えなくなる障害が発生した時、復旧までの加速器待機時間と電気代から被害額を算出した。例えば 6 月に RIBF ウラン加速の実験中に PM12:00 から次の日の PM12:00 まで 24 時間停止したと仮定すると、障害復旧までの加速器待機の為の電気代は 1 日数百万円[†]にもなる。これはシステムを冗長化する為サーバ 1 台を増設する費用の数十倍になる、と考えられる。

3. 冗長化すべきサービス

3.1 NFS

理研 RIBF 制御で採用されている Linux IOC の種類は次に挙げる 3 種類である。CAMAC クレートコントローラーである東陽テクニカ CC/NET[1]、PC Engines 社 x86 組込みボード[2]、そして組込み EPICS で運用を行っている横河電機 F3RP61-2L である[3]。これらはメンテナンス性の観点から EPICS 基本プログラム (以下 base)とデータベースを一つのサーバで集約管理し、そのサーバ上で NFS サービスを走らせて全ての Linux IOC がファイルシステムを共有している[4]。また CA クライアント用アプリケーションサーバのユーザ領域(/home 等) も NFS サーバに格納して、データを共有している。この為 NFS サービスが停止すると上記システム全体が停止するという重大な問題点が考えられる。

3.2 FTP

電磁石電源制御に、組込みシステム向けリアルタイム OS である VxWorks を搭載した IOC は複数採用されている[4]。それらはカーネルと EPICS base, データベースを FTP サーバ一台からダウンロードする事で、運用されていた。したがって Linux IOC と同じく FTP サービスが止まると全ての VxWorks IOC も停止する。

3.3 Zlog

RIBF では KEK で開発された Zlog システムが利用されている[5]。Zlog とは、Zope プロダクトと PostgreSQL データベースと Log Monitor Server から構成されるオペレーション・ログシステムである。RIBF 制御系では開発者が意図した日々の運転ログノートの電子化というよりは、むしろ、ある過去の時刻における加速パラメータの検索、マシンタイム終了後のログの解析と言う目的に用いられている場合が多い。仮にこのシステムが停止してもマシンタイムは続行できるが、ビームや加速器のコンディションやビーム解析の重要な情報を失う事になる為冗長化必須のサービスである。ただし Web で閲覧機能を提供する Zope プロダクトと Zope に関しては冗長化を省いた。なぜなら最低限 PostgreSQL と Log Monitor Server が走っていればオペレーションの情

a-uchi@riken.jp
† 燃料調整費は無視している

報を最低限収集する事が可能だからである。

3.4 EPICS CA クライアント

EPICS コラボレーションから提供されている GUI 構築ツールキットである MEDM/EDM でクライアントが数多く開発されている。上記以外では C 言語, Java, Shell script で開発されたアプリケーションを使用して加速器オペレーションを行っているが、いずれも 1 台のアプリケーションサーバに集約して開発運用を行っていた。オペレーション時における具体的なクライアント立ち上げ方法は次に挙げる手順である。

1. クライアント PC で接続スクリプト実行
2. 呼び出された SSH クライアントで当該アプリケーションサーバにログイン
3. CA クライアントを起動
4. 自身 PC の X サーバに GUI 表示

以上一連の流れを見ると、加速器オペレーションを行うのに必要なサービスは SSH である。しかし稼働中のサーバで SSH だけ落ちるという事は考えにくく、経験からサーバ全体に問題(ハードウェアの故障を含む) を抱えているケースが多い。よってサービスだけでなくサーバ全体を冗長する仕組みが必要である。さらに 28GHz SCECR イオン源のログやマニュアル管理に Wiki ベースのシステムを導入している事から[6]HTTP も冗長すべきサービスである。

3.5 DNS, LDAP

加速器制御ネットワークはセキュリティの観点から理研所内 LAN とは完全に独立した設計になっている。よってホスト名と IP アドレスの解決の為に DNS (Domain Name System)サーバを内部で運用している。また複数 Linux サーバに共通しているログイン名とパスワードの認証は LDAP (Lightweight Directory Access Protocol)で管理している。この２つに関してもサービスが止まると全てのサーバにログインできない、ホスト名でアクセスする事ができない、といった大規模な障害が起こる事が予想される。

4. 冗長化手法

4.1 フェイルオーバークラスタ

3 で述べたサービスの内 NFS, FTP, Zlog (PostgreSQL と Log Monitor Server)に関しては Linux 上にフェイルオーバークラスタを構築する事で冗長化した。フェイルオーバークラスタとは、サーバに障害が発生した時に代替サーバが処理を引き継ぐ事で実現する、高可用性システムの事である。本システムでは Linux-HA プロジェクトの成果物であるオープンソースの Heartbeat[7]を採用した。フェイルオーバークラスタの導入について KEK から Rose Datasystems 社 RoseHA を利用した報告[8]がされているが、我々はソフトウェアライセンスのコスト面での優位性を重視しオープンソースソフトウェアを積極的に利用した。Heartbeat は主系と待機系がお互いを死活監視し、主系の障害を感知した際にフェイルオーバーが起き、サービスが待機系に引き継がれる仕組みになっている。図 1 に RIBF 制御系で実装された NFS、Zlog クラスタ、図 2 に FTP クラスタの概要を示す。

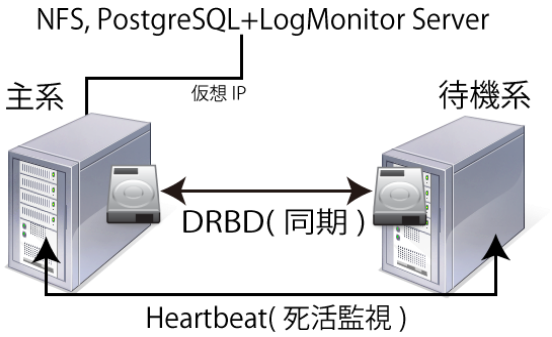

図 1 : NFS, Zlog クラスタ概要

NFS クラスタと Zlog クラスタは、DRBD (Distributed Replicated Block Device)[9]を用いて主系と待機系の共有ディスクに割り当てたブロックをネットワーク越しに自動同期させている。通常動作時は主系が共有ディスクをマウントしてデータをそこへ格納している。Heartbeat が障害を検知しフェイルオーバーが起こった際は、主系が共有ディスクを切り離し、待機系が代わって共有ディスクをマウントする事で主系が格納したデータを引き継いでサービスを続行する、という仕組みである。

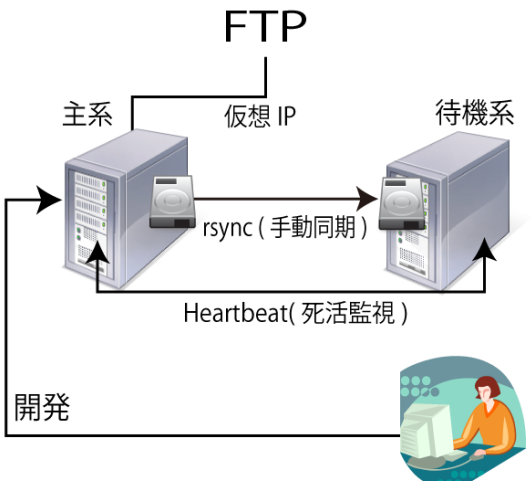

図 2: FTP クラスタ概要

一方で FTP クラスタに関しては DRBD ではなく rsync[10]を用いる事で主系と待機系間のデータ同期を行った。DRBD ではなく rsync を用いた理由の一つは、リアルタイムにデータを同期せずとも運用に問題が無い為である。なぜならフェイルオーバー時サービスに関するデータを引き継がずに数秒 FTP 接

続が切れたとしても、VxWorks IOC は問題なくシステムを稼動し続ける事ができたからである。また主系と待機系が共有しているデータは EPICS base とデータベースになるが、すでに完成されたシステムな為更新頻度は少なく、自動ではなく手動でファイル同期させたかった事も理由の一つである。これは更新前の状態を待機系に残しておき、更新に伴うバグがあった際は一つ前のバージョンに迅速に戻せる仕組みを意図した。システムスペックを表 1 に示す。

表 1：サービス別システム仕様

種類	OS	主サービス	サーバ CPU
NFS	Scientific Linux 4.3	nfsd, DRBD-0.7	AMD Opteron 2212HE
FTP	CentOS 4.2	vsftpd-2.0.1, rsync	Intel Xeon 3050
Zlog	CentOS 4.7	PostgreSQL 8.0.1, DRBD-0.7, Log Monitor Server (at KEK)	Intel Xeon L5410

4.2 DNS ラウンドロビン

ローコストに EPICS CA クライアント用サーバの冗長化を行う為に DNS ラウンドロビンを用いた。一般的に言う DNS ラウンドロビンとは、DNS の設定で一つのホスト名に複数 IP アドレスを割り当てる事で行う負荷分散技術の事である。だが DNS は障害を起こしているサーバの IP アドレスを返す場合もあるので、DNS ラウンドロビンは負荷分散にはなるが冗長化にはならない、と我々は考えていた。一方で死活監視とアクセスを振り分ける仕組みを同時に持たせるにはロードバランサを導入すれば解決するが、それはコスト増大の要因となる。そこで Microsoft Internet Explorer7 や Firefox3 といった最近の Web ブラウザが、接続ホスト名に対して障害が起きた IP アドレスを返した場合でもタイムアウト後に正常なサーバへ接続しなおす、という振る舞いをする事に我々は注目した。以上の事を踏まえると、Apache を利用した Web サービスに関しては DNS ラウンドロビンで冗長化可能と考えると同時に、もし SSH クライアントに関しても上記 Web ブラウザと同じ振る舞いをするのであれば我々の要求を満たすに足りると考え、テストを行った。CentOS5.3 上で行った結果、SSH クライアントはコネクションできないサーバの IP アドレスを振られたとしてもタイムアウト後他方にリトライして接続する、という振る舞いをする事を確認した(図 3)。ただデフォルトの設定ではタイムアウトまでに約 3 分かかってしまい、これでは実用的ではないので /etc/ssh/ssh_config の Connection Timeout の値を変更する事で対応した。Microsoft Windows XP 上にインストールした Cygwin[11]でもテストを行ったが、こちらも同様の振る舞いをした。以上より最近の各種 Linux ディストリビューションにパッケージされている OpenSSH は対応可能と思われる。

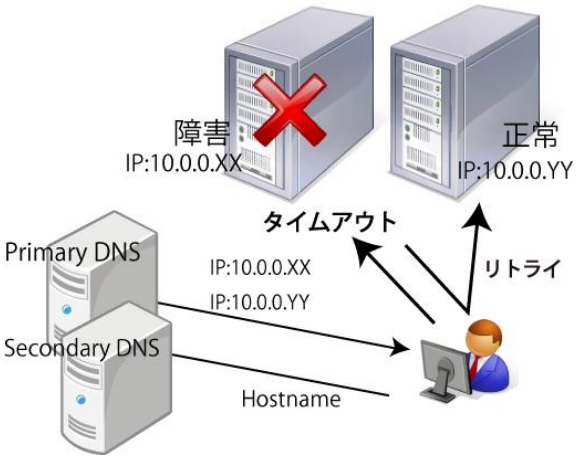

図 3: DNS ラウンドロビンを用いた冗長化概要

4.3 二重化

DNS と LDAP に関しては Primary/Secondary サーバ二台で運用する事にした。二重化はプロバイダ、インターネット回線でごく一般的に利用されている手法である。本システムでは BIND[12] と OpenLDAP[13]を同じハードウェアにインストールし、二台構成とした。

5. 障害事例

1. 2008 年 8 月主系 FTP サーバが原因不明でシャットダウンする障害が発生。待機系にサービスが切り替わって VxWorks IOC は運用し続ける事ができた。
2. 2009 年 3 月 NFS クラスタの待機系にメモリ故障が発生。待機系なので特に問題は起こらずにシャットダウン後修理完了。
3. 2009 年 10 月 Primary DNS のネットワークカード故障による障害が発生。代わりに Secondary DNS が名前を解決する事で対応可能と思われた。しかしながら、当然ではあるがソースに直接ホスト名を記入してあるプログラムは Primary にアクセスしてタイムアウト後に Secondary にアクセスするという振る舞いをする。動的アクセスを伴うシステムでは上記が頻発する為、システム全体が重くなってしまう結果となり、影響は小さいとは言えなかった。以上により HTTP 等の静的アクセスにおける Secondary DNS は大きな意味があるが、動的アクセスするシステムにおける Primary/Secondary 構成の DNS は今後改善が必要である、と考えられる。

4. 2010 年 5 月主系 Zlog サーバでハードウェア故障。システムリブートの必要があったが、強制的にフェイルオーバーさせる事で対応し、システム運用の面では問題はなかった。

6. まとめ・結語

RIBF 制御系における各種サービス別の冗長化手法について詳細に報告した。大型実験施設の様な２４時間動き続ける施設では冗長化システムの導入は必須である、と考えられる。障害がなければ日の目を見る機会は無いが、万が一の場合はシステム停止時間を最小限に抑える事ができる。またオープンソースソフトウェアを利用する事でローコストにシステム構築をする事に成功した。

導入の際入念な障害テストと構築面・運用面共に技術が要求され、特にフェイルオーバークラスタに関しては RoseHA の導入事例報告と比較して簡便なソリューションにはなっているとは言い難い。導入のしきいを下げる為にもオープンソースを利用した構築例のさらなる報告を願う。

7. 謝辞

RIBF 電力における資料を提供して頂いた理研仁科センター 藤縄雅氏に感謝致します。

参考文献

[1] M. Komiyama et al., “CONTROL SYSTEM FOR THE RIKEN RI-BEAM FACTORY” Proc. 3 Annu Meet. Particle Accelerator Society of Japan and 31 Linear Accelerator Meet, Japan, Sendai, **P.932** (2006)
[2] A. Uchiyama et al., “DEVELOPMENT OF EMBEDDED SYSTEM FOR RUNNING EPICS IOC BY USING LINUX AND SINGLE BOARD COMPUTER” Proc. of ICALEPCS07, Knoxville, Tennessee, USA, 2007, **P.334**
[3] M. Komiyama et al., “UPGRADING THE CONTROL SYSTEM OF RIKEN RI BEAM FACTORY FOR NEW INJECTOR USING EMBEDDED EPICS” Proc. 6 Annu Meet. Particle Accelerator Society of Japan, Japan, Tokai, **P.432** (2009)
[4] M. Komiyama et al., “STATUS OF CONTROL SYSTEM FOR RIKEN RI-BEAM FACTORY” Proc. of ICALEPCS07, Knoxville, Tennessee, USA, 2007, **P.189**
[5] K. Yoshii et al., Proc. ICALEPCS07, Knoxville, Tennessee, USA, 2007, **P.299**
[6] A. Uchiyama et al., “Construction of Client System for 28GHz SC-ECRIS” RIKEN Accel. Prog. Rep. 43 (2009), accepted
[7] http://www.linux-ha.org/
[8] S. Kusano et al., “High Availability Cluster System using Linux-PC” Proc. 2 Annu Meet. Particle Accelerator Society of Japan and 30 Linear Accelerator Meet, Japan, Tosu, **P.456** (2005)
[9] http://oss.linbit.com/drbd/
[10] http://www.samba.org/rsync/
[11] http://www.cygwin.com/
[12] http://www.isc.org/software/bind
[13] http://www.openldap.org/